UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/073,851	02/11/2002	Grzegorz J. Czajkowski	004-7056	8021

22120 7590 10/28/2005

ZAGORIN O'BRIEN GRAHAM LLP
7600B N. CAPITAL OF TEXAS HWY.
SUITE 350
AUSTIN, TX 78731

EXAMINER	
BANANKHAH, MAJID A	
ART UNIT	**PAPER NUMBER**
2195	

DATE MAILED: 10/28/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/073,851	CZAJKOWSKI ET AL.	
	Examiner	Art Unit	
	Majid A. Banankhah	2195	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *24 August 2005*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *26-49* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *26-49* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

Response to Amendment and arguments

1. This office action is in response to amendment and remarks filed on August 24, 2005. Applicant's amendments necessitated the new grounds of rejection. Claims 26-49 have been considered for examination.

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

3. Claims 33-41 are rejected under 35 U.S.C. 103(a) as being unpatentable over **Blaukopf et al.** (U.S. Patent Application Publication US 2002/0095521) in view of **Holmberg** et al. (U.S. Patent No. US 6,345,351, hereinafter Holmberg);

As to independent claim 33:

A method for sharing one or more physical memory spaces (which will be explained) between a first and second process, the method comprising:

maintaining analogous memory address ranges between the first and second processes, wherein one of the first and second processes executes native code and the other process executes safe language code [e.g., see "first application is written in a platform independent language" and associated discussion §0016, and "second application 200 written in the processor's native code" and associated discussion §0016, see also discussion §0025: "Each such stream connection has its own memory buffers, also **Blaukopf** teaches the first address range of the first process and the second address range of the second process both map to a common physical memory area, see memory buffers §0025 and physical memory disclosure §0012, and see first application §0017];

However, **Blaukopf** does not *explicitly* teach the following limitations:

Holmberg teaches the address ranges in the first process and the analogous address ranges in the second process are mapped to same portions of the one or more physical memory space [**Holmberg,** col. 4, lines 29-53, memory map causes the first, a first memory map causes the first job to access a first shared physical page in the memory whenever the first job accesses any location in a first virtual page, and second memory map ... and the associate discussion]. It would have been obvious to one of ordinary skill in the art at the time the invention was made to improve upon the system taught by **Blaukopf** by implementing the improvements detailed above because it would provide **Blaukopf's** system with the enhanced capability of avoiding the need to copy the speculatively executed job's data from the temporary storage area into the shared memory at the time of commitment and therefore, reduce additional overhead that slows down the overall information processing rate [see **Holmberg**, col. 5, lines 11-17]. This is a more efficient use of the available buffer space and reduces the probability that a data packet will be delayed due to congestion in the queue buffers.

As to the dependent claim 34:
the method of claim 33 wherein the maintaining comprises:
the first process reserving a first address range and requesting a buffer from the second process [**Holmberg**, col. 4, lines 39-43 a first memory map causes the first job to access a first shared physical page in the memory... and associated discussion];

responsive to the request from the first process, the second process allocating a second address range analogous to the first address range [**Holmberg**, col. 4, lines 43-46, a second memory map also causes the second job to access the first shared physical page in the shared memory whenever the second job accesses any location in the first virtual page ..., and associated discussion].

As to the dependent claim 35:
The method of claim 34 further comprising:
the second process creating a buffer object and mapping the second address range to a first portion of the one or more physical memory spaces and communicating [**Holmberg**, col. 7, lines 35-45, also, **Blaukopf**, see "first application is written in a platform independent

language" and associated discussion §0016; see also discussion §0025: "Each such stream connection has its own memory buffers ...", and "first application is written in a platform independent language" and associated discussion §0016; see also discussion §0025: "Each such stream connection has its own memory buffers ..."] a buffer object identifier and a physical memory space identifier that identifies the first physical memory space portion to the first process; [**Holmberg**, col. 7, lines 54-60, the memory controller generate memory access cycle, and associated discussion, also **Blaukopf** e.g., see §0017 "The first application 100 may pass a function call to the second application 200 through the first 120 and second 220 mediation modules, respectively; see cont'd discussion §0019, "Communication between mediation modules occurs using a stream protocol ..."; see cont'd discussion §0025: "Each such stream connection has its own memory buffers ...";];

the first process mapping the first address range to the first physical memory space portion as identified by the physical memory space identifier [**Holmberg**, col. 5, lines 39-48, a first memory map causes the first job,...and associated discussion, also **Blaukopf** see memory buffers §0025 and physical memory disclosure §0012].

As to the dependent claim 36:
the first process communicating the buffer object identifier to a native code caller [**Blaukopf,** e.g., see §0017 "The first application 100 may pass a function call to the second application 200 through the first 120 and second 220 mediation modules, respectively; see cont'd discussion §0019, "Communication between mediation modules occurs using a stream protocol ..."; see cont'd discussion §0025: "Each such stream connection has its own memory buffers ...",].

As to the dependent claim 37:
the physical address ranges comprise a set of one or more physical memory pages [**Holmberg**, e.g. see, shared physical page and shared virtual page and associated discussion, col. 5, lines 34-58].

As to the dependent claim 38:
allowing at least two of the virtual address ranges in the first process to overlap [**Holmberg**, e.g., see col. 5, lines 18-24, In yet another aspect, the second memory map comprises a shared page table and a private page table, and associated discussion].

As to the dependent claim 39:

The method of claim 38 further comprising:
the first process requesting an address range (A1, A1+S1) that overlaps with a previously allocated address range (M, M+S2), wherein A1 and M represent address in the first process and S1 and S2 represent memory space sizes [**Holmberg**, e.g., see col. 8, lines 18-29, virtual and physical address range and an N-bit offset address may be used to ..., and associated discussion];
the second process allocating an address range (M', M'+S2), wherein A1' and M' represent addresses in the second process [**Holmberg**, e.g., see col. 8, lines 18-29, virtual and physical address range and an N-bit offset address may be used to ..., and associated discussion], mapping (M', M' + (A1+S1-M)) in the second process to a same first portion of the one or more physical memory spaces to which (M, A1+S1) in the first process is mapped [**Holmberg**, e.g., see col. 5, lines 18-23, using the second memory map to cause the second job to access the first shared physical page in the shared memory whenever the second job accesses any location in the first virtual page], and mapping (M'+(A1+S1-M), M'+S2) in the second process to a same second portion of the one or more physical memory spaces to which (A1+S1, M+S2) in the first process is mapped [**Holmberg**, e.g., see col. 5, lines 39-45, a first memory map causes the first job to access a first shared physical page in the memory whenever the first job accesses any location in a first virtual page in the set of shared virtual pages, wherein the first shared physical page is in the set of shared physical pages].

As to the dependent claim 40:
40. (New) The method of claim 33 further comprising maintaining a list that indicates the address ranges, wherein the list allows detection of at least one of overlapping address ranges and nested address ranges [**Holmberg**, e.g., see col. 5, lines 18-24, In yet another aspect, the second memory map comprises a shared page table and a private page table, and associated discussion].

As to the dependent claim 41:
at least one of the first process address ranges and the second process address ranges comprise virtual addresses [**Holmberg**, e.g., see col. 5].

4. Claims 42-46, 26-32 and 47-49 are rejected under 35 U.S.C. 103(a) as being unpatentable over **Blaukopf et al.** (U.S. Patent Application Publication US 2002/0095521), in view of **Chaney** et al.

(European Patent Application EP 0 841 617 A2 hereinafter **Chaney**); further in view of Holmberg et al. (U.S. Patent No. US 6,345,351, hereinafter **Holmberg**);

As to independent claim 42:
A computer program encoded on one or more computer readable medium the computer program comprising:
a first language code in a safe language code and executing the first language code, [e.g., see "first application is written in a platform independent language" and associated discussion §0016], executable to map the first environment address range to a physical memory space [**Blaukopf** teaches mapping the first address range to a physical memory area identified by the first identifier, see first application §0017];

a second language code in a native language code and executing the second language code, [e.g., see "second application 200 written in the processor's native code" and associated discussion §0016, see also discussion §0025: "Each such stream connection has its own memory buffers], executable to map the second environment address range to a physical memory area [**Blaukopf** teaches the first address range of the first process and the second address range of the second process both map to a common physical memory area [see memory buffers §0025 and physical memory disclosure §0012];

However, **Blaukopf** does not *explicitly* teach the following limitations:

Allocate an address range in the first environment in response to a request for a buffer, and allocate an address range in a second environment, which execute the second language code.

Chaney teaches allocating a memory address range for each of the first memory buffer object and the second memory buffer object, in a second process, the second process executing native code [e.g., a technical feature of the invention is to use a single buffer for both request buffer space and response buffer space, see col. 2, lines 41-48] .

It would have been obvious to one of ordinary skill in the art at the time the invention was made to improve upon the system taught by **Blaukopf** by implementing the improvements detailed above because it would provide **Blaukopf's** system with the enhanced capability of "a

system that makes more efficient use of the available buffer space and reduces the probability that a data packet will be delayed due to congestion in the queue buffers" [e.g., see **Chaney**, col. 1, lines 32-34].

Additionally, **Blaukopf** does not *explicitly* teach the following limitations:

Holmberg teaches mapping the first address range and the second address range to the same (**shared**) physical memory space [**Holmberg,** col. 4, lines 29-53, memory map causes the first, a first memory map causes the first job to access a first shared physical page in the memory whenever the first job accesses any location in a first virtual page, and second memory map ... and the associate discussion]. It would have been obvious to one of ordinary skill in the art at the time the invention was made to improve upon the system taught by **Blaukopf** by implementing the improvements detailed above because it would provide **Blaukopf's** system with the enhanced capability of avoiding the need to copy the speculatively executed job's data from the temporary storage area into the shared memory at the time of commitment and therefore, reduce additional overhead that slows down the overall information processing rate [see **Holmberg**, col. 5, lines 11-17]. This is a more efficient use of the available buffer space and reduces the probability that a data packet will be delayed due to congestion in the queue buffers.

As to dependent claim 43:
an interface code to handle communications between the first and second environments [**Blaukopf,** e.g., see §0017 "The first application 100 may pass a function call to the second application 200 through the first 120 and second 220 mediation modules, respectively; see cont'd discussion §0019, "Communication between mediation modules occurs using a stream protocol ..."; see cont'd discussion §0025: "Each such stream connection has its own memory buffers ..."].

As to dependent claim 44:
the interface code is implemented according to the Java native interface, the safe language code is implemented according to the Java programming language, and the one of the first and second environments that executes the safe language code comprises a virtual machine [see **Holmberg**, col. 10, lines 8-21, For example, a Java

Virtual Machine (JVM) is an emulator that allows Java byte code to execute on almost any processor. Emulator programs may either directly interface with the underlying processing equipment, and associated discussion].

As to dependent claim 45:
the other of the first and second environments that executes native code comprises an operating system [**Blaukopf,** e.g., see §0016 "the first mediation module 120 may issue a command causing the operating system to produce an instance of the second application 200 and the second mediation module 220].

As to dependent claim 46:
the first environment address range being correspondent to the second environment rage comprises the address ranges being a same size [see **Holmberg**, col. 8, lines 15-17, Instead, each computer program sees a virtual address space that may or may not be the same size as that of the physical address space, and associated discussion].

As to independent claim 26:
A method for handling sharing of a physical memory space (which will be explained below) between a first process and a second process, the method comprising:
A first process executing native code of a program [e.g., see **Blaukopf** "first application is written in a platform independent language" and associated discussion §0016], a second process executing safe language code of the program [e.g., see "second application 200 written in the processor's native code" and associated discussion §0016, see also discussion §0025: "Each such stream connection has its own memory buffers],

Blaukopf does not *explicitly* teach the following limitations:

Chaney teaches allocating a first address range in the first process and a second address range in the second process
[e.g., a technical feature of the invention is to use a single buffer for both request buffer space and response buffer space, see col. 2, lines 41-48] .

It would have been obvious to one of ordinary skill in the art at the time the invention was made to improve upon the system taught by **Blaukopf** by implementing the improvements detailed above because

it would provide **Blaukopf's** system with the enhanced capability of "a system that makes more efficient use of the available buffer space and reduces the probability that a data packet will be delayed due to congestion in the queue buffers" [e.g., see **Chaney**, col. 1, lines 32-34].

Additionally, **Blaukopf** does not *explicitly* teach the following limitations:

Holmberg teaches mapping the first address range and the second address range to the same (**shared**) physical memory space [**Holmberg,** col. 4, lines 29-53, memory map causes the first, a first memory map causes the first job to access a first shared physical page in the memory whenever the first job accesses any location in a first virtual page, and second memory map ... and the associate disscussion]. It would have been obvious to one of ordinary skill in the art at the time the invention was made to improve upon the system taught by **Blaukopf** by implementing the improvements detailed above because it would provide **Blaukopf's** system with the enhanced capability of avoiding the need to copy the speculatively executed job's data from the temporary storage area into the shared memory at the time of commitment and therefore, reduce additional overhead that slows down the overall information processing rate [see **Holmberg**, col. 5, lines 11-17]. This is a more efficient use of the available buffer space and reduces the probability that a data packet will be delayed due to congestion in the queue buffers.

As to dependent claim 27:
at least one of the first and second address ranges comprises virtual addresses [**Holmberg**, e.g., see col. 5].

As to dependent claim 28:
the physical memory spaces comprise a set of one or more Physical pages [**Holmberg**, e.g. see, shared physical page and shared virtual page and associated discussion, col. 5, lines 34-58].

As to dependent claim 29:
the allocating is responsive to the first process requesting a direct buffer for the first address range from the second process [**Blaukopf**, e.g., see "memory buffers" discussion §0025, see §0017 "The first application 100 may pass a function call to the second application 200 through the first 120 and second 220 mediation modules, respectively; see cont'd discussion §0019, "Communication between mediation

modules occurs using a stream protocol ..."; see cont'd discussion §0025: "Each such stream connection has its own memory buffers ...";].

As to dependent claim 30:
the second process, causing generation of a direct buffer object and an associated direct buffer object identifier; and
communicating the direct buffer object identifier and a physical memory space identifier for the shared physical memory space to the first process [**Holmberg**, col. 7, lines 54-60, the memory controller generate memory access cycle, and associated discussion, also **Blaukopf** e.g., see §0017 "The first application 100 may pass a function call to the second application 200 through the first 120 and second 220 mediation modules, respectively; see cont'd discussion §0019, "Communication between mediation modules occurs using a stream protocol ..."; see cont'd discussion §0025: "Each such stream connection has its own memory buffers ...";].

As to dependent claim 31:
communicating the direct buffer object identifier and the physical memory space identifier to other processes that execute native code [**Holmberg**, col. 7, lines 35-45, also, **Blaukopf**, see "first application is written in a platform independent language" and associated discussion §0016; see also discussion §0025: "Each such stream connection has its own memory buffers ...", and "first application is written in a platform independent language" and associated discussion §0016; see also discussion §0025: "Each such stream connection has its own memory buffers ...", also **Holmberg**, col. 7, lines 54-60, the memory controller generate memory access cycle, and associated discussion, also **Blaukopf** e.g., see §0017 "The first application 100 may pass a function call to the second application 200 through the first 120 and second 220 mediation modules, respectively; see cont'd discussion §0019, "Communication between mediation modules occurs using a stream protocol ..."; see cont'd discussion §0025: "Each such stream connection has its own memory buffers ...", and **Holmberg**, col. 5, lines 39-48, a first memory map causes the first job,...and associated discussion, also **Blaukopf** see memory buffers §0025 and physical memory disclosure §0012].

As to dependent claim 32:
maintaining an encoding that indicates at least one of overlapping address ranges and nested address ranges within one of the first and second processes [**Holmberg**, e.g., see col. 5, lines 18-24, In yet

another aspect, the second memory map comprises a shared page table and a private page table, and associated discussion].

As per independent claim 47:
An apparatus comprising:
A memory [see for e.g. **Blaukopf**, a memory and associated discussion §0016, see also discussion §0025]; and
A first process and second process executes native code and the other the first and second process Executes safe language code [e.g., see "first application is written in a platform independent language" and associated discussion §0016, and see "second application 200 written in the processor's native code" and associated discussion §0016, see also discussion §0025:];

Blaukopf does not *explicitly* teach the following limitations:

Chaney teaches allocating an address range in a first process for a first code and an analogous address range in a second process for a second code [e.g., a technical feature of the invention is to use a single buffer for both request buffer space and response buffer space, see col. 2, lines 41-48].

It would have been obvious to one of ordinary skill in the art at the time the invention was made to improve upon the system taught by **Blaukopf** by implementing the improvements detailed above because it would provide **Blaukopf's** system with the enhanced capability of "a system that makes more efficient use of the available buffer space and reduces the probability that a data packet will be delayed due to congestion in the queue buffers" [e.g., see **Chaney**, col. 1, lines 32-34];

Additionally **Blaukopf** does not *explicitly* teach the following limitations:

Holmberg teaches sharing of at least a portion of the memory between the first code and the second code [**Holmberg,** col. 4, lines 29-53, memory map causes the first, a first memory map causes the first job to access a first shared physical page in the memory whenever the first job accesses any location in a first virtual page, and second memory map ... and the associate discussion]. It would have been obvious to one of ordinary skill in the art at the time the invention was made to improve upon the system taught by **Blaukopf** by implementing the improvements detailed above because it would

provide **Blaukopf's** system with the enhanced capability of avoiding the need to copy the speculatively executed job's data from the temporary storage area into the shared memory at the time of commitment and therefore, reduce additional overhead that slows down the overall information processing rate [see **Holmberg**, col. 5, lines 11-17]. This is a more efficient use of the available buffer space and reduces the probability that a data packet will be delayed due to congestion in the queue buffers.

As to dependent claim 48:
at least one of the first process address range and the second process address range comprises virtual addresses [**Holmberg**, e.g., see col. 5].

As to dependent claim 49:
means for allowing detection of at least one of nested address ranges and overlapping address ranges[**Holmberg**, e.g., see col. 5, lines 18-24, In yet another aspect, the second memory map comprises a shared page table and a private page table, and associated discussion].

5. **Obviousness-type double patenting Rejection:**
The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. See In re Goodman, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); In re Longi, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); In re Van Ornum, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); In re Vogel, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); and In re Thorington,418 F.2d 528, 163 USPQ 644 (CCPA 1969).

> "Double patenting rejection of application claims was fully justified where applicant, in course of expanding first application to disclose enough more by way of details, alternatives, and additional uses to support broad, dominating, generic claims in later applications, has disclosed no additional invention or discovery other than that what was already claimed in patent on first application; there is significant difference between justifying broadening of claims and disclosing additional inventions." In re Van Ornum, 214 USPQ 761 (CCPA 1982).

6. Claims 1-25 is rejected under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 1 - 24 of parent application 09/841,719, now U.S. Patent 6,834,391 (**Czajkowski et al.**) in view of Holmberg et al. (U.S. Patent No. US 6,345,351, hereinafter **Holmberg**).

Although the conflicting claims are not identical, they are not patentably distinct from each other because of corresponding language that recites virtually all of the same elements and functions claimed in the previously patented invention, e.g., "a first process", "a second process", "native code", "platform independent code", and, in particular, inter-process communications between a process executing native code and a second process executing platform independent code, except claims of **Czajkowski et al.** does not *explicitly* teach of "mapping the first address range and the second address range to the shared physical memory space.
However, the claimed differences would be obvious to a programmer of ordinary skill because the instant claims are merely broader and/or alternate variations of the claims recited in the
parent application.

Because the instant claims merely eliminate and/or alternately claim limitations from the set of elements and functions claimed in the parent application, such modifications would be readily apparent to a programmer of ordinary skill.

For example, **Holmberg** teaches the address ranges in the first process and the analogous address ranges in the second process are mapped to the shared memory portions of the one or more physical memory space [**Holmberg,** col. 4, lines 29-53, memory map causes the first, a first memory map causes the first job to access a first shared physical page in the memory whenever the first job accesses any location in a first virtual page, and second memory map ... and the associate discussion]. It would have been obvious to one of ordinary skill in the art at the time the invention was made to improve upon the system taught by **Czajkowski,** by implementing the improvements detailed above because it would provide **Czajkowski** system with the enhanced capability of avoiding the need to copy the speculatively executed job's data from the temporary storage area into the shared memory at the time of commitment and therefore, reduce additional overhead that slows down the overall information processing rate [see **Holmberg**, col. 5, lines 11-17]. This is a more efficient use of the

available buffer space and reduces the probability that a data packet will be delayed due to congestion in the queue buffers.

Terminal Disclaimer

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent is shown to be commonly owned with this application. See 37 CFR 1.130(b). Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

For post GATT applications, (i.e., applications filed after June 8, 1995), the rule § 1.321 (4) (c) (3) requires a provision that must be included. The following requirement is UNCHANGED by GATT and therefore a terminal disclaimer is required for the instant application, i.e., *"shall be enforceable only for and during such period that said patent is commonly owned with the application or patent which formed the basis for the rejection."*

7. **Prior Art not relied upon:**

Please refer to the references listed on the attached PTO-892, which is relied upon in the claim rejections detailed above.

8. Applicant's amendment necessitated the new grounds of rejection. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 C.F.R. § 1.136(a).

A SHORTENED STATUTORY PERIOD FOR RESPONSE TO THIS FINAL ACTION IS SET TO EXPIRE THREE MONTHS FROM THE DATE OF THIS ACTION. IN THE EVENT A FIRST RESPONSE IS FILED WITHIN TWO MONTHS OF THE MAILING DATE OF THIS FINAL ACTION AND THE ADVISORY ACTION IS NOT MAILED UNTIL AFTER THE END OF THE THREE-MONTH SHORTENED STATUTORY PERIOD, THEN THE SHORTENED STATUTORY PERIOD WILL EXPIRE ON THE DATE THE ADVISORY ACTION IS MAILED, AND ANY EXTENSION FEE PURSUANT TO 37 C.F.R. § 1.136(a) WILL BE CALCULATED FROM THE MAILING DATE OF THE The application has been amended as follows:
ADVISORY ACTION. IN NO EVENT WILL THE STATUTORY PERIOD FOR RESPONSE EXPIRE LATER THAN SIX MONTHS FROM THE DATE OF THIS FINAL.

How to Contact the Examiner:

9. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Majid Banankhah, whose telephone number is 571-272-3770. A voice mail service is also available at this number. The Examiner can normally be reached on Monday, and Wednesday - Friday, 7:00 AM - 3:30 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, An Meng-Al who can be reached on 571-272-3756. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

All responses sent by U.S. Mail should be mailed to:

Commissioner for Patents
PO Box 1450
Alexandria, VA 22313-1450

PTO CENTRAL FAX NUMBER:
703-872-9306

- Any inquiry of a general nature or relating to the status of this application should be directed to the TC 2100 Group receptionist: **(703) 305-3900**.

MAJID BANANKHAH
PRIMARY EXAMINER